0 / 516470
03 DEC 2004

PCT/GB2003/002708

The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP10 8QQ

REC'D 31 JUL 2003
WIPO PCT

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., plc, P.L.C. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN COMPLIANCE WITH RULE 17.1(a) OR (b)

Signed

Dated 24 July 2003

24JUN02 E727929-1 D02917
P01/7700 0.00-0214444.2

THE PATENT OFFICE
G
21 JUN 2002
LONDON

Patents Form 1/77
Patents Act 1977
(Rule 16)

The Patent Office

Request for grant of a patent

The Patent Office
Cardiff Road
Newport
South Wales NP10 8QQ

1. Your reference
   5435701/AM

2. Patent Application Number
   0214444.2   21 JUN 2002

3. Full name, address and postcode of the or of each applicant *(underline all surnames)*

   Thirdspace Living Limited
   1st Floor
   Alcatel Building
   Voyager Place
   Shoppenhangers Road
   Maidenhead
   Berkshire
   SL6 2PJ

   Patents ADP number *(if known)* 8119133002

   If the applicant is a corporate body, give the country/state of its incorporation
   Country: **ENGLAND**
   State:

4. Title of the invention

   USER INTERFACE SYSTEM

5. Name of agent
   Beresford & Co

   "Address for Service" in the United Kingdom to which all correspondence should be sent
   2/5 Warwick Court
   High Holborn
   London WC1R 5DH

   Patents ADP number 1826001

6. Priority details

Country	Priority application number	Date of filing

Patents Form 1/77

7. If this application is divided or otherwise derived from an earlier UK application give details

Number of earlier application | Date of filing

8. Is a statement of inventorship and or right to grant of a patent required in support of this request?

YES

9. Enter the number of sheets for any of the following items you are filing with this form.

Continuation sheets of this form	
Description	58
Claim(s)	
Abstract	
Drawing(s)	9 + 1

10. If you are also filing any of the following, state how many against each item.

Priority documents	
Translations of priority documents	
Statement of inventorship and right to grant of a patent *(Patents form 7/77)*	1 + 2 COPIES
Request for preliminary examination and search *(Patents Form 9/77)*	
Request for Substantive Examination *(Patents Form 10/77)*	
Any other documents *(please specify)*	

11. I/We request the grant of a patent on the basis of this application

Signature Beresford & Co
BERESFORD & Co

Date 21 June 2002

12. Name and daytime telephone number of person to contact in the United Kingdom

ALAN JOHN SHAW MACDOUGALL

Tel: 020 7831 2290

USER INTERFACE SYSTEM

The present invention relates to an interactive television system. The invention has particular although not exclusive relevance to an interactive television system having personal video recorder functionality.

Conventionally, television programmes have been broadcast to users via RF signals transmitted from terrestrial base stations, via signals transmitted from overhead satellites and via signals transmitted over cable to user premises. Each of these systems offers the user the ability to watch a number of different channels which can be selected by the user. These existing systems, however, require all of the channels to be transmitted to the user's television receiver, which then tunes into and displays one of the channels in accordance with the user's selection. In some of these conventional systems, the user must subscribe to the service provider in order to be able to view some of the channels. However, since each user's television receiver receives all of the channels, users can still gain access to restricted channels using appropriate hacking equipment which can bypass the service provider's security.

Further, with these conventional systems, the television viewing experience for the user is one in which the user is effectively passive. In other words, the programme schedule is fixed in advance by the service providers and the only choice that the user has is which channel he

wishes to view. New interactive television systems are beginning to emerge in which the user can interact through the television with the service providers to control the content that is delivered, thereby creating a more personal entertainment experience. These systems employ menu-based user interface systems to allow the user access to the various services that are available. However, to date, these menu-based interfaces are difficult and confusing for the user to operate. Further, current menu interface systems are designed as "one size fits all" systems, typically transmitting to and displaying for every user of the system in a particular region the same channel line-up (usually in numerical order) and programming information in the same format and style.

In order to provide users having conventional television sets with the ability to be able to interact with the service providers, a user set top box (STB) is provided. At present, various service providers have produced their own set top box, each having different hardware and software loaded thereon. The service providers have focussed on adding significant processing power to the set top box and download proprietary software for maintaining, processing and displaying the bulk of the control data such as, for example, user profile data, programme guide data and usage data. As a result of the complexity of these proprietary set top boxes, the [illegible]

significant user support. In particular, each time a change is made to such systems, each user's set top box needs to be checked and upgraded or even replaced. Further, with this type of system the development of new applications is more difficult and time consuming, since each application must be written in a format that suits the processor speed, operating system and internal architecture of each set top box.

Additionally, some of these set top boxes include a hard disc for recording programmes or videos for subsequent playout. These recording systems also require additional software control for controlling the storage and retrieval of the files to and from the hard disc. However, since the software for controlling access to the video files stored in the hard disc is provided locally within the set top box, it is difficult to implement any parental control over who has access to the stored content.

Another proposal for providing personal video recording capabilities is to provide storage remotely within a video server which is coupled to the set top box through a data network. With this system, however, users in the same geographical area are likely to want to access content stored within the remote video server at similar times, thereby leading to significant increases in the amount of traffic in the data network at some times of the day.

The present invention aims to provide an alternative interactive television system having personal video recorder capabilities in which local storage is provided within a user's set top box but in which the access to the stored content is controlled by a remote server coupled to the set top box.

Other aspects and features of the present invention will become apparent from the following detailed description of preferred embodiments which is given with reference to the accompanying drawings in which:

Figure 1 is a schematic block diagram of the architecture of a system for providing a user with access to a plurality of services and content;

Figure 2a is a schematic block diagram illustrating the main components of a user set top box forming part of the system shown in Figure 1 and Figure 2b illustrates the format of a typical user request;

Figure 3 is a schematic block diagram illustrating the main components of a user interface server forming part of the system shown in Figure 1;

Figure 4 is a block diagram illustrating the main components of an application server forming part of the system shown in Figure 1

components of a database forming part of the system shown in Figure 1;

Figure 6 is a functional flowchart illustrating the way in which the user can access the user interface menu system which provides the user with access to the services and content provided by the remote servers shown in Figure 1;

Figure 7 is a schematic diagram illustrating the main components of a remote control shown in Figure 1;

Figure 8 schematically illustrates the form and layout of a main menu page showing a Yourspace option, a Videospace option, a TVspace option and an Openspace option; and

Figure 9 is a schematic block diagram of the architecture of an alternative system for providing a user with access to a plurality of services and content.

OVERVIEW

Figure 1 is a schematic block diagram illustrating the main components of a system 1 which allows users to gain access to a plurality of services and content from a plurality of remote servers. The different users of the system 1 access the services and content via a respective user device 3, three of which are shown in Figure 1 and referenced 3-1, 3-2 and 3-3. As shown in Figure 1, in this embodiment, each user device 3 includes a television

5, a set top box (STB) 7, a remote control device 9 and a keyboard 11. Menus for accessing the various services and content that are available are displayed to the user on the television 5 and the user selects and controls the accessing of the services and content using the remote control 9 and/or the keyboard 11.

In this embodiment, the services that the user can access include:

i) video on demand (e.g. films on demand, music on demand, time shifted TV, personal video recorder, video commerce etc.) from a video server 15 and a video database 17;

ii) e-mail from a mail server 19 which is connected to the Internet via a firewall 20-1;

iii) an electronic programme guide (EPG) from an EPG server 21;

iv) electronic commerce from a shopping server 23;

v) Internet/world wide web access via a web server 25 and a fire wall 20-2;

vi) broadcast TV (BTV) including basic television channels, premium channels, pay-per-view etc. provided by a BTV server 27 and a BTV receiver 28; and

vii) user services such as billing information, user profiles etc. provided by a management and billing server 29.

As shown in Figure 1, in this embodiment, the accessing of the services or content provided by the application servers 30 is carried out via a number of user interface servers 31, three of which are shown in Figure 1 and referenced 31-1, 31-2 and 31-3. The user interface servers 31 are operable to receive user requests transmitted from the associated set top box 7 via an IP data network 33 and a load balancer 35 (which shares the user requests between the user interface servers 31, based on how busy each is). In this embodiment, the user gains access to the different services and content provided by the application servers 30 via menu pages of a graphical user interface. In this embodiment, these menu pages are generated by the user interface server 31 and downloaded over the IP data network 33 as HTML (hypertext markup language) files to the set top box 7. A web browser (not shown) in the set top box 7 then generates or renders the appropriate menu page from the received HTML file, which it displays to the user on the television 5.

When a user makes a selection from a menu page (using the remote control 9 or the keyboard 11) an appropriate user request is generated by the user set top box 7 and transmitted back to the user interface server 31. In response, the user interface server 31 tries to generate the next menu page itself from data stored in local caches (not shown). If the data is not available locally, then the user interface server 31 passes the user request on to the appropriate application server 30

which obtains the appropriate data and passes it back to the user interface server 31. The user interface server 31 then uses the received data to generate a personalised HTML file which it transmits back to the user set top box 7.

The data necessary for generating the various menu pages and the various user profile data are stored centrally within a database 39 which can be accessed by any of the application servers 30 or the user interface servers 31.

In this embodiment, the services or content of each application server 30 are accessed by the users from menu pages generated by the user interface servers 31. However, the resulting services or content may be delivered directly from the application servers 30 to the user or they may be delivered through the user interface server 31. In this embodiment, the application servers 30 which transmit large amounts of data to the users transmit their data directly to the users via the IP data network 33. These application servers 30 include the video server 15, the web server 25 and the broadcast TV server 27. The other servers (i.e. the mail server 19, the EPG server 21 and the shopping server 23) return their services through the user interface servers 31.

As those skilled in the art will appreciate, since the user interface servers 31 generate the menu pages used

user interface servers 31 can ensure a common "look and feel" to the menu pages regardless of the application server 30 being accessed. As a result, the user interface menu system of this embodiment is easier to understand, use and learn than those of the prior art systems available today. Further, the user interface servers 31 use intelligent caching techniques and user profile information to personalise in an efficient way the menu pages downloaded to each user.

One of the novel features of this embodiment is the provision of personal video recorder (PVR) capabilities within the set top boxes 7 which are effectively controlled by the application servers 30 and the user interface servers 31. In particular, in this embodiment, the set top boxes 7 include a hard disc (not shown) for recording selected videos and/or television programmes. The selection of the content to be recorded is controlled by the user via the menu pages or automatically on the basis of predictions of what the system believes the user would like to watch, determined from user profile data collected and maintained by the management and billing server 29. Further, in this embodiment, video streams may also be stored for each user within the video database 17 associated with the video server 15. When the user wants to watch a video or television programme that has been recorded in their personal video recorder, they navigate through the menu pages to retrieve a user specific PVR menu page identifying the content that is currently stored in their personal video recorder. Each

of the items listed in the PVR menu page includes a link identifying where the content is stored, either locally within the user's set top box 7 or remotely within the video database 17. In this way, if the user selects an item from their PVR menu page, the appropriate content can be retrieved.

A brief description has been given above of the way in which users access services and content provided by a number of different application servers 30. A more detailed description will now be given of some of the components used in the system 1 shown in Figure 1 and in particular in relation to their operation to provide personal video recorder (PVR) services to the users.

Set Top Box

Figure 2a is a functional block diagram illustrating the main components of one of the set top boxes 7 shown in Figure 1. As shown in Figure 2a, the set top box 7 includes a network interface unit 201 which operates to interface the set top box 7 to the IP data network 33. HTML files received from a user interface server 31 over the IP data network 33 are passed, through the network interface unit 201, to a web browser 203 which processes the HTML file to generate a menu page which it outputs to a frame buffer 205 for display on the television 5. The web browser 203 is also operable to receive user input either from the remote control 9 via the remote [illegible]

example, to scroll through options on the currently displayed menu page and/or to select options from the current menu page. The HTML file received from the user interface server 31 also includes links for other menu pages and/or services and content that are available from the current menu page. The HTML file received from the user interface server 31 also includes instructions for the web browser which associates key presses on the remote control 9 and/or the keyboard 11 to these links. When the user presses a key on the remote control 9 and/or the keyboard 11, the web browser 203 then interprets this key press based on the received instructions to identify the link that the user has selected. In this embodiment, these instructions are Javascript instructions and the web browser 203 includes an appropriate Javascript command processor (not shown) for interpreting the instructions. The web browser 203 then generates an appropriate user request for transmission to a user interface server 31, which user request includes the link corresponding to the key press together with user data (such as user ID, session ID etc.) stored in the user data memory 211.

Figure 2b schematically illustrates the information in a typical user request 215 transmitted from the set top box 7 to the user interface server 31. As shown, the user request 215 includes:

i) a source IP address 221 identifying the IP address of the set top box 7 that transmitted the request;
ii) a destination address 223 (in this embodiment, the

URL address of the user interface server 31) identifying that the request is to be transmitted through the IP data network 33 to a user interface server 31;

iii) a current user ID 225 which identifies the current user that is watching and interacting with the user device 3;

iv) a session ID 227 identifying a current user session to which the transmitted user request 215 relates; and

v) an application identifier 229 and a screen identifier 231 which together form the above-mentioned link associated with the current menu page being displayed, which identifies to the user interface server 31 the application server 30 to which the request should be transmitted and the particular menu page or service or content requested by the user.

The set top box 7 also includes a video player 213 (such as an MPEG decoder) which operates under control of the web browser 203. In particular, the web browser 203 can control the video player 213 to request a particular video stream from the video server 15 or a particular television channel from the broadcast television server 27. In this embodiment, these user requests 215 are passed to the network interface unit 201 which then forwards them over the IP data network 33 to a user [illegible]

mentioned above, in this embodiment, the video server 15 and the BTV server 27 are arranged to stream the requested video or television channel directly to the user over the IP data network 33. The stream of video or television channel data received from the IP data network 33 is then passed through the network interface unit 201 to the video player 213. In accordance with instructions from the web browser 203, the video player 215 then either processes the received video or television channel data or it stores it unprocessed within a local hard disc 214 used to provide personal video recorder capabilities. As those skilled in the art will appreciate, since the video player 215 does not process received video or television channel data which is to be stored in the hard disc 214, it can receive multiple streams for different videos and/or television programmes and store these separately in the hard disc 214. This is possible, since each data packet received from the IP data network 33 will include an identifier identifying to which stream the packet belongs. Additionally, since the received video or television channel data is not for viewing, the data can be "trickle-fed" to the set top box 7 at a reduced data rate than would be required for streams of video or television programmes that are to be viewed in real time. In this embodiment, personalised user adverts are also preferably downloaded with the video or television programme to be recorded which can then be inserted within the video or television programme at any time during playout.

If the received stream of video or television channel data is not to be stored in the hard disc 214, then the video player 215 processes the data (which will typically be encoded using, for example, MPEG) to regenerate the frames of the video or television channel, which it then passes back to the web browser 203. The web browser 203 then outputs the received video or television channel frames to the frame buffer 205 for display on the television 5. In this embodiment, the web browser 203 can control the size of the video or television channel frames displayed to the user on the television 5 so that, for example, the video or television channel is displayed to the user in a portion of the television screen, with the remainder of the screen being used to display menu options that are available.

In the event that the user selects an item to be viewed from their personal video recorder menu page, the web browser 203 uses the link associated with the selected item to instruct the video player 213 to retrieve the file either from the video server 15 or from the hard disc 214. If the requested content is stored within the video database 17, then the video server 15 retrieves the appropriate video file and starts streaming the video file to the user set top box 7 over the IP data network 33. If, however, the requested content is stored locally within the hard disc 214, then the video player 213 retrieves and decodes the appropriate content file and [illegible]

USER INTERFACE SERVER

Figure 3 is a functional block diagram illustrating the main components of a user interface server 31. As discussed above, the user interface server 31 is arranged to generate HTML files describing personalised menu pages to be transmitted to user set top boxes 7 in response to received user requests 215. Each user interface server 31 tries to generate these HTML files itself without having to pass the request 215 to the appropriate application server 30, in order to try to minimise the processing burden on the application servers 30 and on the database 39. The user interface server 31 is also responsible for carrying out common processing functions (such as user login, error handling etc.) which are required by two or more of the application servers 30.

As shown in Figure 3, the user interface server 31 includes an interface unit 301 which is operable to interface the user interface server 31 to the IP data network 33 and to the load balancer 35. The interface unit 301 is operable to receive messages from the load balancer 35 and to pass them to a listening unit 303. The listening unit 303 is arranged to listen for user requests 215 transmitted from the user set top boxes 7 and to pass these to a request handling unit 305. The request handling unit 305 is responsible for validating the user making the request and for checking that the request is a valid one. The request handling unit 305 also checks to see if the user request can be dealt with by the user interface server 31 from data stored in an

HTML cache 309-1 or an XML cache 309-2. The contents of these and other caches will be described in more detail later. If the request handling unit 305 determines that the necessary information for responding to the user request 215 is stored within the HTML or the XML caches 309, then the request handling unit 305 passes the cached information directly to a response handling unit 307 which uses this cached information to generate a personalised HTML file which it outputs back to the user set top box 7 via the interface unit 301 and the IP data network 33.

If the request handling unit 305 determines that the user interface server 31 cannot respond directly to the user request 215, then it determines which application server 30 the request 215 is to be directed and then retrieves appropriate user information from a user data cache 310 that will be required by that application server 30 to respond to the user request 215. At the same time, the request handling unit 305 also determines if any common functions (such as user login) are to be performed and if so, it instructs a common functions processor 311 to carry out the appropriate common function and return the result. The request handling unit 305 then passes the original user request 215 together with the additional user information retrieved by the request handling unit 305 to the appropriate application server 30 via an interface unit 313. After the application server 30 has [illegible]

application server 30 returns an XML (extended markup language) file which identifies the information to be displayed to the user. As those skilled in the art will appreciate, an XML file only describes the information that is to be displayed, it does not describe how it should be displayed (i.e. the format and layout). This formatting information is provided, in this embodiment, by style sheets (not shown), some of which are stored in an XSLT cache 309-3, with the remainder being stored in a hard disk 315.

In this embodiment, when the response handling unit 307 receives an XML file from one of the application servers 30, it stores the XML file in the XML cache 309-2 and combines it with the appropriate style sheet from the XSLT cache 309-3 or the hard disk 315, to generate an HTML file which is stored in the HTML cache 309-1. The response handling unit 307 then uses data from the user data cache 310 to personalise the HTML file which it sends back to the appropriate user set top box 7 via the interface unit 301 and the IP data network 33. As shown in Figure 3, the response handling unit 307 can also invoke the common functions performed by the common functions processor 311. This therefore allows application servers 30 to be able to trigger one of the common functions, such as the user login function by returning an appropriate instruction to the response handling unit 307.

In this embodiment, one of the key goals of the user

interface server 31 is to try to reduce the number of user requests 215 that are passed on to the application servers 30 by, wherever possible, responding to the user's request 215 using data from caches within the interface server 31. As discussed above, the user interface server 31 includes a user data cache 310, an HTML cache 309-1, an XML cache 309-2 and an XSLT cache 309-3. The type of data stored in each of these caches will now be explained.

User Data Cache

The user data cache 310 stores all of the user information that is available in the database 39 for each user of the system 1. In this embodiment, this equates to approximately 500 bytes of data for each user. This data includes, among others, the user name, user age, user login name, user PIN (personal identification number), user set top box type, session ID, user login status bit, user subscription level, user family name, user set top box ID, current television channel or video programme being viewed, user E-mail address, user language, user background colour and other user preferences.

HTML Cache

The HTML cache 309-1 caches various HTML files that define the content and layout of menu pages. In this embodiment, there are essentially two different types of

describe menu pages or parts of menu pages which are the same for all users. For example, the HTML file describing the initial menu screen that shows the various options that are available will be common to all users (except for minor user personalisations which can be made when the file is about to be downloaded to the user). The dynamic HTML files that are stored in the HTML cache 309-1 are generated by the response handling unit 307 from an XML file received from, for example, an application server 30. These dynamic HTML files therefore describe menu pages showing menu data which may be specific to the particular user (for example illustrating the favourites of that user). In this embodiment, each dynamic HTML file is cached for a predetermined period of time (such as 500 seconds) defined by an application server 30.

XML Cache

The XML cache 309-2 stores XML files either generated from the common functions processor 311 or from the application servers 30. As mentioned above, XML files define the information that will be displayed in a menu screen but not the layout of that information on the menu screen. For example, the XML file may identify the programme listings for a selected TV channel for today. Since this information is likely to be requested by other users, the user interface server 31 caches this XML file in the XML cache 309-2. In this way, for example, if another user wishes the same information but requires a different style sheet to generate the HTML file (because,

for example, they have a different type of set top box 7 or television 5), then the user interface server 31 does not have to obtain the same XML file from the application server 30 again. It can simply retrieve the XML file from the XML cache 309-2 and transform it into the appropriate HTML file using the appropriate style sheet for the other user.

XSLT Cache

In this embodiment, the XSLT cache 309-3 stores the style sheets which are used to generate the HTML files from the XML files. In this embodiment, there are five different classes of style sheet -

i) a "form" style sheet in which one or more text boxes are provided for allowing the user to input text;

ii) a "carousel" style sheet in which the user can scroll through menu options;

iii) a "short electronic programme guide" style sheet which is used to provide programme information relating to a current television channel to the user;

iv) a "bill" style sheet which is used to provide detailed billing information to the user; and

v) a "mail" style sheet which is used to provide E-mail information to the user.

In this embodiment, there are several different style [illegible]

widescreen/narrowscreen, PAL/NTSC etc.). As those skilled in the art will appreciate, the task of combining the XML file with the style sheet to generate the HTML file can be relatively time-consuming (of the order of 200 milliseconds). In this embodiment, the style sheets are stored within the XSLT cache 309-3 in a pre-processed format which makes them easier to combine with the XML file.

Intelligent Caching

One of the possible problems with using caching techniques such as those used in the user interface server 31 is the possible storage requirements if user-specific HTML pages are to be cached. In particular, a system operating with U users, each caching P pages of size S will require storage of order U x P x S, which will grow large as the number of users, number of pages or complexity of each page increases.

In order to minimise the storage requirements, in this embodiment, an intelligent caching technique is used which distinguishes, for any given HTML file, which content is static (i.e. common to all users) and thus only needs to be stored once and which content is dynamic (or user-specific) and hence needs to be stored on a per-user basis. In this embodiment, this is achieved by providing delimiters in the style sheets which indicate which sections of their output are static and which are dynamic. The caching then proceeds as follows:

i) The first time a user requests a specific page

(such as the programme listing for a given channel on a given day), the generated HTML page will be processed and separated into its static and dynamic portions using the delimiters inserted into the style sheet.

ii) The static portions are then stored in a static data store within the HTML cache 309-1 and the dynamic portions for the particular user are stored in a dynamic data store within the HTML cache 309-1.

iii) When a second user requests the same page, the HTML file will again be generated but only the user-specific dynamic portions will be stored in the HTML cache 309-1 - the static portions will be the same as those stored during the request of the first user and therefore do not need to be stored.

iv) When a user who has already requested the page requests it again, the cached page will be reconstructed by combining the static portion and the user-specific dynamic portion, thus recreating the user's page from the HTML cache 309-1, without the entire contents of the page being stored for each individual user.

Variable Swapping

In order to improve the efficiency of the caching system used in this embodiment, the user interface server 31 supports a technique known as post-parse interpolation [illegible]

customisations such as a change in background colour or the addition of the user name to the menu screen, to be applied to the HTML file after it has been generated using the style sheet and the XML file. It is the use of this variable swapping technique that allows the system to be able to store static HTML files for all users whilst at the same time being able to personalise the HTML files for individual users at serve time (i.e. at the time it is downloaded to the user). In this embodiment, this is achieved by the response handling unit 307 which swaps specific user data into the HTML file at the time that it is about to be downloaded to the user set top box 7, using a variable swapping algorithm. These variables are referred to as hash-hash variables because they are represented in the style sheet as ##variable_name##, and are swapped using an efficient process that is much faster than the style sheet/XML transformation process.

Whenever a substitution of this sort is to be made, a placeholder of the form ##variable_name## is inserted into the style sheet or XML file so that it appears in the resulting HTML file. At serve time, the corresponding variable for the user is retrieved from the user data cache 310 and inserted into the HTML file.

In this embodiment, the same variable swapping technique is also used to swap in machine constants associated with the user interface server 31, into generic XML files received from the application servers 30. In particular,

since the application servers 30 can transmit XML files to different user interface servers 31, they use placeholders within the XML file to identify constants that are specific to the user interface server 31. When the response handling unit 307 receives a generic XML file having such a placeholder, it swaps in the appropriate constants that are specific to that user interface server 31. For example, the XML file may refer to a particular icon that is to be downloaded with the menu page to the user set top box 7. The directory location that this icon is stored may be different on each of the user interface servers 31. Therefore, by inserting the name of the icon within the ## delimiters, the response handling unit 307 can replace the icon name with the correct storage location for the icon on that user interface server 31.

Application Server

The application servers 30 receive user queries from the user interface servers 31 together with user details and information generated from any common functions which have been required to action the request. The application server 30 operates to deliver the user's requested service or menu page by processing the received request and data and by retrieving data relevant to the request from the database 39. In order to ensure optimum performance in the system 1 and to meet the goal of limiting the queries on the database 39, the application

Figure 4 is a schematic block diagram illustrating the main components of one of the application servers 30. This block diagram has been shown in general form so that it is applicable to all of the application servers 30. As shown, the application server 30 includes a UIS interface unit 601 for interfacing the application server 30 to the user interface servers 31. The UIS interface unit 601 is operable to receive user requests 215 together with the added user information provided by the user interface servers 31 which it passes to an application request handling unit 603. The application request handling unit 603 processes the received data to determine: (i) if the request should be rejected; (ii) if the user request can be responded to from data stored in a results cache 605; or (iii) if the user request should be forwarded to an application processor 607. In particular, in this embodiment, the application request handling unit 603 checks to ensure that each user request that it receives is for that application server 30. It does this by checking the application identifier 229 forming part of the user request 215 with the application identifier associated with that application server 30. If these identifiers are different, then the application request handling unit 603 rejects the user request and returns an appropriate error code back to the user interface server 31.

As mentioned above, the application servers 30 generate XML files that describe the information to be inserted within a menu page. These XML files are designed to be

generic in nature so that they can be processed by any of the user interface servers 31 and so that they can be used for servicing user requests received from other users. In this embodiment, the XML files generated for previous user requests are stored for a predetermined period of time in the results cache 605. Therefore, when the application request handling unit 603 receives a valid user request, it checks the XML files stored in the results cache 605 to determine if the XML file for responding to the user request is stored in this cache. If it is, then the application request handling unit 603 retrieves the XML file from the results cache 605 and returns it to the user interface server 31 that transmitted the user request. The application request handling unit 603 also informs the user interface server 31 that this XML file is cachable and for how long it is cachable. The XML file is also returned together with data identifying the user who made the request. In this embodiment, the application request handling unit 603 also passes the more generic XML files that are generated to the other user interface servers 31, also indicating that it is cachable and for how long it is cachable, so that these other user interface servers 31 can update their XML caches 309-2 accordingly.

If the application request handling unit 603 determines that it cannot service the user request from previously generated XML files stored in the results cache 605, the [illegible]

user interface server 31 to the application processor 607. In this embodiment, it is the application processor 607 that determines what service and/or what menu page the user is requesting. The application processor 607 does this using the screen identifier 231 forming part of the received user request 215 and data stored within a menu logic and data store 609. In particular, the menu logic and data store 609 stores data associated with each possible screen identifier which defines the information to be displayed in the next menu page together with menu logic defining what user selections can be made on that page. Therefore, when the application processor 607 receives a user request, it identifies the screen identifier 231 forming part of the received user request 215 and it retrieves the appropriate data and menu logic from the store 609. The application processor 607 then processes the retrieved data and the user data received with the request to determine what information it needs to respond to the request and to determine if it needs to retrieve any of that information from the database 39. If the application processor 607 determines that it does need to query the database 39, then it first checks a database (DB) cache 611 and a generic query cache 613 which store results of previous requests for data sent to the database 39. If the required information is not stored in these caches, then the application processor 607 formats an appropriate database query and outputs it to the database 39 via a database interface unit 615. When the application processor 607 receives the raw database data (such as the user favourites table) back

from the database 39, it stores it in the DB cache 611. The application processor 607 then processes the returned database data to obtain the requested information (such as the favourites of a particular user) in a format suitable for returning to the user, which it stores in the generic query cache 613.

In this embodiment, the database cache 611 stores the data that is most frequently used by the application server 30 and is refreshed on a regular basis or when triggered by the database 39. When the data in the database cache 611 is updated in this way, the application processor 607 also reprocesses the data in order to refresh the data within the generic query cache 613. In this way, the data within these caches can be kept up to date for responding to subsequently received user requests.

After the application processor 607 has obtained the relevant information for responding to the user request, it passes the information together with the appropriate menu logic (defining allowed user selections and links therefor etc.) back to the application request handling unit 603. The application request handling unit 603 then packages the information and the menu logic into an XML file which it stores in the results cache 605 and returns to the user interface server 31 in the manner discussed above.

Management and Billing Server

Whilst the management and billing server 29 conforms with the above generic description, it is worth discussing in more detail its purpose within the system 1. In particular, the management and billing server 29 is operable to provide various user services such as user billing and user profiling. In this embodiment, the management and billing server 29 is also responsible for initially logging a user onto the system 1 and setting up the various user profiles and user tables within the database 39 for the new user. During this initial logon procedure, the user will provide the management and billing server 29 with details such as the user's age, password, E-mail addresses, spending limits, user name, world wide web home page, search page, user language, country etc. The management and billing server 29 is then responsible for creating the necessary user tables within the database 39 which in turn triggers the updating of the user data within the various caches within the system 1, in order to accommodate the new user.

The management and billing server 29 is also responsible for tracking payment of bills by the different users and for blocking the provision of services or content to users if they do not make payment.

In this embodiment, users can access the data maintained by the management and billing server 29 via the user interface server, for example, to identify what the

outstanding amount owed by that user is or to identify the different films that have been purchased by that user in the current billing period.

In order to carry out the billing, the management and billing server 29 reads the user billing table (not shown) from the database 39, where all the application servers 30 write their transactions identifying what services and content have been delivered to the various users. The management and billing server 29 then calculates the appropriate amount for those services or content and adds it to the user's bill.

In this embodiment, the management and billing server 29 also monitors the different user requests that are received by the user interface servers 31 which are stored within the user request log 411 shown in Figure 4. The management and billing server 29 then uses this information in order to determine user profiles for the different users of the system 1. For example, the management and billing server 29 can perform various statistical processings on the requests made by each user in order to try to identify the types of television programme or video films that the user likes. This user profile information can then be stored in the database 39 and used, for example, by the electronic programme guide server 21. In particular, the EPG server 21 may use this user profile information to make suggestions to [illegible]

Video Server

As discussed above, one of the functions of the video server 15 is to provide personal video recorder services to each of the users. In particular, in this embodiment, it is the video server 15 which controls the downloading and storing of selected video streams and television programmes either into the hard disc 213 in the user's set top box 7 or in the video database 17. The video server 15 is also responsible for controlling each user's ability to access the stored content by providing and maintaining within the database 39, the data required to generate each user's PVR menu page.

In operation, when the video server 15 receives a request to record a video or television programme that is to be broadcast by the broadcast TV server 27, the video server 15 transmits a request to the broadcast TV server 27 to output a copy of the requested programme to the video server 15. The video server 15 then stores the broadcast programme in the video database 17 and then updates the user's PVR list with the file name and storage location. Alternatively, the video server 15 may store the requested programme into the hard disc 214 within the user's set top box 7. In this case, because the user wishes to record the programme, it is not essential to stream the video data over the data network 33 so that it can be watched in real time on the user's television 5. Instead, the programme can be "trickle-fed" down through the data network 33 and stored within the hard disc 214 of the set top box 7. The video server 15 then

records the file name and location of the downloaded file within the user's PVR list maintained in the database 39. Alternatively, the user may wish to record a television programme as it is being broadcast directly into the user's set top box. In this case, the video server 15 simply controls the timing when the video player 213 in the set top box 7 starts and stops recording.

One advantage of performing the recording at the video server 15 is that the video server 15 can insert personalised adverts into the content based on the profile for the user who requested the content. This advert may be a passive advert or it may be the home page of an interactive advert that allows the user to select it and then spend time browsing within subpages of the advert. These pages may be stored together with the advert or they may be kept as a link to an appropriate web site on the web server 25. Another advantage of performing the recording at the video server 15 is that the recorded video can be processed in order to extract temporal tagging data which can be used to provide an index file. This index file can then be transmitted to the set top box 7 together with the original content file and can be used to allow accessing of the stored file at any location therein (i.e. in a random access fashion) and other motion control effects such as fast-forward or rewind playout. The index file can also be used to prevent the user from skipping advertisements embedded [illegible]

user can skip adverts.

Subsequently, when the video server 15 receives a request from a user to view their content stored in their personal video recorder, the video server 15 retrieves the data necessary for generating the user's PVR menu page. This data may be cached within the video server 15 (or in the user interface server 31) or the video server 15 may have to retrieve the data from the database 39. The video server 15 then passes the appropriate XML file defining the user's PVR menu page back to the user interface server 31. This XML file will identify the different items of content that is stored in the user's personal video recorder together with data identifying where each item is stored. The user interface server 31 then generates the appropriate PVR menu page and returns it to the user's set top box 7 which outputs the PVR menu page to the user on the television 5. As those skilled in the art will appreciate, since the video server 15 maintains different lists of PVR data for each user, it does not matter if several users are associated with the same set top box, each of those users will have their own content in their own PVR.

One advantage of this embodiment is that because control over when recording is to be started and stopped is carried out by the video server 15, any last minute updates to the electronic programme guide can be taken into account. For example, if a user wishes to record a programme starting at 1700 hours and ending at 1730

hours but the previous programme overruns by 10 minutes, then since the video server 15 uses the centrally stored EPG data which is almost always kept up-to-date, the video server 15 can ensure that only the desired programme is recorded. This is one of the advantageous features of the PVR system of this embodiment, since in a conventional set top box personal video recorder system, the electronic programme guide which is used to control the recording of the programme is typically downloaded some time in advance of the event playing out on the system. Therefore, either extra time must be recorded either side of the allocated time slot(which requires more storage space) or some of the programme may be missed.

Another advantage of this embodiment is that the user profiles determined and managed by the management and billing server 29 can be used to accurately suggest programmes for recording for each user. Further, in this embodiment, each user can identify from the EPG listing which programmes are their favourites and the management and billing server 29 can use this favourites information to control suggestions that are made to each user. Further, different levels of suggestion may be provided to cater for predictions of "hot favourites" which the system predicts that the user will definitely want to watch and, for example, "interesting favourites" which the system thinks that the user may like to watch. [illegible]

[illegible]

than in multiple locations in the individual set top boxes 7, a more sophisticated and expensive profiling system may be used which can avoid some of the problems associated with existing personal video recorder systems. For example, many set top box systems will record all broadcasts of a television show even if they are repeats. However, by using a suitably programmed profile engine, this can be detected to ensure that repeats are not recorded.

Further, since the profiling engine is provided centrally, it can use user-identified favourites to control the recording of programmes and prioritise the recordings for the user. Further, since the user profile information stored within the management and billing server 29 includes user details such as the type of programme each user likes to watch (e.g. science fiction, drama etc.), the management and billing server 29 can perform an analysis of the viewing habits of the different users to be able to identify programmes that appeal to users who like, for example, science fiction. This information can then be used to respond to queries such as: "I like science fiction, what does everyone else who likes science fiction watch?".

Another advantage of this server-controlled PVR architecture is that the server side of the system can convert non-supported content in the video server 15 prior to downloading and storage in the user's set top box 7. This converting can take the form of changes in

bit rate, changes in video codec, changes in audio codec, changes in encryption technique, addition of supplementary data (such as language-specific subtitles) etc.

In this embodiment, the user can navigate through the menu pages in order to access a menu page which shows the number and types of recordings that are currently stored in the user's personal video recorder. For example, this menu page might show a graphical illustration showing unused space, space used for recordings which have been marked permanent, space taken up by "hot suggestions" made by the management and billing server 29 and identifying recordings which have not been watched. If the user receives a message informing them that their personal video recorder is full, the user can then use this menu page to see how the storage space is being used. The video server 15 also provides, in this embodiment, menu pages for allowing the user to delete content from their personal video recorder and to define rules for the automatic deletion and maintenance of the user's personal video recorder. In this way, the user can define, for example, that any content that has been recorded on the basis of a suggestion by the management and billing server 29 may be deleted to make room for a user-selected recording and/or that no content should be deleted unless it has been watched etc.

services is to ensure that the system can provide some level of parental control so that, for example, minors under the age of eighteen are not able to access adult material. In this embodiment, this is achieved automatically through the filtering on a user-by-user basis of a full electronic programme guide (EPG) listing. In particular, since each user must log into the system before they can gain access to any of the content provided by the application servers 30, and since each user has an associated user profile (which defines, among other things, the user's age) stored centrally within the database 39, the EPG server 21 can use this information to filter out content which is not suitable for the user requesting the information. Therefore, for example, minors under the age of eighteen using the system do not receive details of any adult programmes in their downloaded EPG listings. Details of these programmes are automatically filtered out from the full EPG listing on a user-by-user basis by the EPG server 21. Consequently, users can only request content that is available to them to be recorded within their personal video recorder.

Additionally, even though an adult and a minor may both be users of the system and may both use the same set top box 7, since the video server 15 effectively generates respective data for each user's PVR menu page, it is possible to ensure that one user of the set top box 7 cannot see and view the content recorded for another user of the same set top box 7. In this embodiment, however, each user can classify each programme or video that they

record into different classes of recordings so that some of the content that they record can be viewed by other users of the set top box 7. In this embodiment, these classes include:

i) shared storage but individual delete - programmes or videos in this class may be recorded by one user of a set top box 7 but may be viewed by any user of the same set top box, however only the user who recorded the programme or video can delete it;

ii) private storage - programmes or videos recorded in this class can only be viewed by the user who recorded them;

iii) private storage but notify other users - programmes or videos stored in this class are not viewable by other users of the same set top box 7 unless another user of the same set top box 7 requests to record the same programme or video, in which case the programme or video is made available to that other user; and

iv) shared storage - programmes or videos recorded in this class can be viewed by any user of the same set top box 7 and can be deleted by any user of the set top box 7.

Database

The database 39 is the single area of the system 1 where all the user's details, transactions and application data are stored. The database 39 is responsible for

database 39 is also responsible for notifying the application servers 30 and the user interface servers 31 when data within the database 39 has changed, so that the internal caches of the servers can be updated.

Figure 5 is a block diagram illustrating the main components in the database 39. As shown, the database 39 includes a server interface unit 701 which operates to interface the database 39 with the application servers 30 and the user interface servers 31. Database queries received from these servers are passed to a database processor 703 which processes data within database tables 705 to respond to the query. As shown in Figure 5, the database tables 705 include a number of application tables 707 which store data relevant for the different application servers 30. For example, these tables store the electronic programme guide data that would be used by the EPG server 21 to generate programme guide listings. The database tables 705 also include user tables 709 which stores the various user information and details used by the application servers 30 and the user interface servers 31. This information includes, for example, the user name, user family name, user status, user login name, user login password, user login PIN, user E-mail address, user favourites, user language, user colour, user country, user PVR list, etc. The database tables 705 also include user detail tables for storing user account information, billing information and details of items purchased etc. Finally, the user database tables 705 also include a set of stored procedures 713

which can be invoked by a request from an application server 30 or a user interface server 31 in order to process some of the data within the database table 705. For example, the stored procedures may be used to process the electronic programme guide which provides programme listings for all channels available from the broadcast TV server 27, to determine the programmes that are playing now on a selection of the TV channels.

In addition to responding to queries received from application servers 30, the database processor 703 is also operable to transmit triggers to the various servers in order to refresh the caches within those servers. In particular, if an application server 30 or one of the user interface servers 31 writes data into the database tables 705, the database processor 703 generates appropriate triggers which it outputs to the other servers within the system 1 so that they can update the relevant parts of their caches. In this way, the database processor 703 can control the synchronisation of the cached data within the system 1.

A description has been given above of a system that allows users to gain access to services and content from a number of remote servers 30 via a user interface server 31. The user gains access to these services and content via a menu-based user interface in which the menu screens are generated within the user interface server 31 and [illegible]

User Menu System

Figure 6 is a functional flowchart illustrating the general operation of the menu-based user interface used in this embodiment. Typically, before a user enters the menu system, they will either be watching a video stream (in step s1) or a broadcast TV programme (in step s3). In order to gain access to the menu system, the user presses (in step s5) a menu key on the remote control 9 or the keyboard 11. In the following description unless otherwise stated, it will be assumed that the user is using the remote control 9 to navigate through the menu system.

Figure 7 schematically illustrates the remote control 9 used in this embodiment. As shown, the remote control 9 includes: a menu key 901, an up key 903, a down key 905, a left key 907, a right key 909 and three function keys 911-1, 911-2 and 911-3. The remote control 9 operates in a conventional way such that if a user presses one of the keys then a corresponding remote control signal 915 will be generated and transmitted to the set top box 7 which receives the signal and determines from it which key was pressed.

Returning to Figure 6, if at step s5 the user presses the menu key 901 while they are watching a video stream or a broadcast TV programme, then the main menu will be displayed in step s7 on the television 5. In practice, what happens in this embodiment is that when the user presses the menu key 901, the set top box 7 generates a

user request for the main menu screen. This request is transmitted to the user interface server 31 which generates the main menu screen from its local caches 309. As discussed above, the user interface server 31 personalises the menu screen for the user (for example in order to add the user's name to the menu screen, to change the background colour and to add the current time etc.) and then transmits it back to the set top box 7 for display on the television 5.

Figure 8 illustrates the format of the main menu 100 used in this embodiment. As shown, the main menu 100 is split into two main parts - a left-hand frame 101 in which various menu categories 107 are presented to the user; and a right-hand frame 103 in which the video or broadcast television programme that the user was watching continues to play. The left-hand frame 101 includes an area at the top of the frame for displaying the name and logo 105 of the service provider that the user is subscribed to (in this case it is the name and logo of Thirdspace). Underneath the logo, there are four menu categories 107-1 to 107-4 to choose from, each having an associated icon 109-1 to 109-4 that is highlighted to identify the category that is currently selected. The left-hand frame 101 also includes an area 111 in which the name of the current user is displayed. Finally, at the bottom of the left-hand frame 101, the current time 113 and date 115 are displayed. In the right-hand frame

upper part 117 and the television programme or video is played out in the central display area 119. In this way, the user can continue to watch the TV programme or video that was playing before the menu key was pressed.

By pressing the up key 903 or the down key 905 on the remote control 9, the user can change the menu category 107 that is currently highlighted. For example, referring to Figure 8 again, the current menu category that is highlighted may be the Videospace category 107-2. If the user presses the up key 903 then the Yourspace menu category 107-1 would become highlighted. Alternatively, if the user had pressed the down key 905 then the TVspace menu category 107-3 would become highlighted. In this embodiment, in order to enter a menu category 107, the user presses the right key 909 on the remote control when that menu category 107 is highlighted. This is illustrated in Figure 6 at step s9. As shown in Figure 6, the result is the displaying of the TVspace menu in step s11, the Videospace menu in step s13, the Yourspace menu in step s15 or the Openspace menu in step s17, depending on which menu category 107 was highlighted at the time.

In this embodiment, the TVspace category 107-3 provides the user with access to the services and content provided by the broadcast television server 27; the Videospace category 107-2 provides the user with access to services provided by the video server 15; the Yourspace category 107-1 provides the user with access to the world wide web

via the web server 25, E-mail via the mail server 19 and their account information via the management and billing server 29; and the Openspace menu category 107-4 provides the user with access to shopping, classified adverts, local information and games via the shopping server 23.

Summary and Advantages

A television-based system has been described which allows users to gain access to a plurality of services and content from a plurality of remote servers. One of the main advantages of the system described above is that the user gains access to the different servers via a common user interface server. With this structure, the system can employ various intelligent caching techniques to reduce the processing burden on the remote servers and on a common database used by the servers. As a result, it is easier to scale the system to operate with more and more users. Further, by generating the menu pages in the user interface server, it is easier to generate a menu-based user interface which has a common look and feel and through which the user can access the services of all of the different application servers. Further, the menu pages can be personalised for each user not just in terms of format but also the content provided to each user.

The system described above provides a user interface that is personalised for each user. The design, selections, content and layout of the screens of the personalised

usage information maintained in the system database. The database is accessed by the user interface server as it processes the user's request for the next or the previous menu screen, for access to a system service or application, or for access to specific content. The user interface server creates a personalised menu screen including design elements, services and content based on the profile data and usage information of the user to which the menu will be presented. Each menu screen presented to a specific user has a consistent design, look and feel and includes services and content targeted to the specific user.

Another advantageous feature of the system described above is the intelligent caching techniques that are employed including the constant swapping techniques which allow generic menu pages to be stored and personalised upon delivery to the individual users. In particular, by using placeholders within the XML documents and style sheets, it is possible to subsequently personalise each menu page by swapping in user specific data for the placeholders. In this way, minor personalisations such as a change in background colour or font, the addition of the user name etc. can be made to the menu page quickly and at the time of delivery. The cached menu pages can therefore be used for a number of different users, thereby saving on cache memory requirements.

A further advantage of the system described above is the use of HTML menu pages and their generation using XML

data and style sheets. In particular, since these are standard formats, it is relatively straightforward for third parties to interface their applications to the system.

Modifications and Alternatives

A detailed description has been given above of a television-based system for allowing users to gain access to television services and media content from a number of remote servers using a graphical user interface displayed on the television. As those skilled in the art will appreciate, various alternatives may be made to the system described above. Some of these modifications and alternatives will now be described.

In the above embodiment, when the user accessed their PVR menu page, it was downloaded from the user interface server 31 to the user's set top box 7 together with data identifying where each item in the PVR menu page is located. This is not essential. Instead, each item in the PVR menu page may be associated with data which identifies the name of the content requested and a link to a central call controller. Such an embodiment is illustrated in Figure 9 which shows a similar system architecture shown in Figure 1 except together with a central call controller 40. In this embodiment, it is the call controller 40 which maintains the details of what content is stored in each user's personal video [illegible]

when a user selects an item from their PVR menu page, the user's set top box 7 transmits a request to the call controller 40 requesting the selected content. The call controller 40 then retrieves the PVR list for the user making the request and determines where the content is stored. The call controller 40 then transmits an appropriate redirect instruction back to the user's set top box 7, redirecting the user's request either to the local hard disc 214 or to the remote video database 17.

The use of such a call controller 40 within the data network provides a number of significant advantages. Firstly, the call controller 40 can keep a log of what content each user wishes to watch from their personal video recorder. This information can then be used to control the deletion of content from each user's personal video recorder based on its utilisation. The use of the call controller 40 is also advantageous in distributed systems where a number of video servers 15 are provided each storing different content. In this case, the call controller 40 can redirect the user to the appropriate video server 15 or to the nearest video server 15 if two video servers 15 have the requested content. Additionally, the use of such a call controller also removes the need to include specific location details in menu pages transmitted over the IP data network 33. Further still, when the call controller 40 receives a request to playout a given content file from the user's personal video recorder, the call controller 40 can build a "playlist" which defines a sequence of files to be

played out, including the user's requested file and, for example, personalised advertisements. The individual content files within this playlist may be stored locally within the hard disc of the user's set top box or remotely in one of the video servers 15.

In the above embodiment, a hard disc was provided in each user's set top box. This was primarily provided for personal video recorder services. Additionally, the hard disc may be used in order to pause live television broadcasts. In effect, if the user presses the pause button during a live television programme, the system can start storing the broadcast television programme in the hard disc for subsequent playout by the user. In practice, a limit must normally be placed on the amount of data that can be recorded in the hard disc in this way. For example, ten per cent of the disc space may be allocated for pausing live television. Depending on the size of the hard disc, this may equate to approximately 30 minutes of recording time.

In the above embodiment, the set top box was connected to the application servers via a user interface server. However, the set top boxes still received content transmitted directly from some of the application servers bypassing the user interface server. This can cause problems if the user interface servers temporarily go off-line. In this case, the set top box will not be able

However, the last logged-in user's profile and menu page are preferably cached within the set top box in order to enable the system to continue playout and capture of incoming live broadcast streams. In this way, the user can still have access to some television services even though connection to the user interface server is temporarily unavailable.

In the above embodiment, the user gained access to the services provided by a plurality of remote servers via a user interface server. This is not essential. For example, the user may gain access to the services or content provided by one or more of the application servers directly, rather than going via the user interface server. The disadvantage of this approach, however, is that if these application servers generate the menu pages and download them directly to the user set top boxes, then it becomes more difficult to maintain a similar look and feel between the menu pages generated by the application servers and the menu pages generated by the user interface server.

In the above embodiment, the user requested services and/or media content from the application servers via the user interface server. In an alternative embodiment, the user may receive menu pages from the user interface server and once a service or media content has been identified for downloading, the user may request that content directly from the appropriate application server. For example, once the user has identified a video to

download from the video server, the user device may direct the request for that video directly to the video server, without it being routed through the user interface server. Such an embodiment has the advantage of reducing the number of requests being handled by the user interface server.

In the above embodiment, a common functions processor was provided in each of the user interface servers. This common functions processor was used to perform functions (such as user login, error handling etc.) that are required by two or more of the application servers. As those skilled in the art will appreciate, it is not essential to provide such a common functions processor. It is also possible to implement functions which may only be required by one of the application servers, especially if it is perceived that the common function will be required by application servers which may be added to the system in the future.

In the above embodiment, the user gained access to the television services and media content using a user set top box and a television. As those skilled in the art will appreciate, it is not essential to use such a set top box and television. For example, the user may gain access to the television services and media content using a personal computer (PC) or the like or a hand-held device such as a personal digital assistant (PDA) or

In the above embodiment, the user interface servers were separate from the application servers. As those skilled in the art will appreciate, one or more of the applications may be run on the same physical machine as the machine running the user interface server. For example, the mail server may be run on the same physical machine as one of the user interface servers. In this case, the user interface server may communicate with the mail server using appropriate memory pointers and call-up routines. Additionally, two or more of the applications may be physically run on a single computer device.

In the above embodiment, the user device is connected to the user interface servers through an IP data network. As those skilled in the art will appreciate, the user device may connect to the user interface server by any appropriate means. For example, the connection may be made via a mobile telephone communication link. Alternatively, the user may connect using a telephone and modem such as an ADSL (asymmetric digital subscriber line) link. Alternatively, the set top box may be connected to the user interface server via a cable or a freespace microwave or optical communication link.

In the above embodiment, a single database was provided which stored details of all of the users subscribed to the system and which was accessed by the different application servers and user interface servers. As those skilled in the art will appreciate, multiple databases may be provided each storing the same information. This

allows database queries from the servers to be shared amongst the different databases. As those skilled in the art will appreciate, such an embodiment would require the databases to be synchronised with each other so that the data stored in each database is the same. Various techniques are known to synchronise multiple databases in this way.

In the above embodiment, the menu pages were generated from HTML web pages downloaded from the user interface server to the user devices. The use of HTML files in this way is preferred since conventional web browser software can be used within the user device to generate the menu page from the received HTML file. Further, menu logic may also be downloaded in the HTML file as Java instructions. This allows the HTML file to contain, for example, details of how a menu carousel should operate, without having to return to the user interface server each time the user scrolls a menu option through a selection window of the carousel. However, as those skilled in the art will appreciate, it is not essential to download the menu pages in HTML format. The pages may be downloaded as images. In this case, when the user presses a key on the remote control or the keyboard, the user device would transmit the appropriate key press to the user interface server which would then interpret the request and download a new image for display. Whilst such an embodiment is possible, it is [illegible]

user device.

In the above embodiment, the menu pages were generated at the server side of the system and downloaded to the user devices. In an alternative embodiment, the user devices may be arranged to generate the menu pages directly from XML files downloaded from the application servers. In such an embodiment, it is not essential to have the user interface servers, since the user devices can then perform the appropriate personalisation of the menu pages. The disadvantage of such an embodiment is that it adds to the complexity of the user devices. Further, if the common functions originally performed by the user interface server are performed in the user device, then this would also increase the vulnerability of the system to hacking by users.

In the above embodiment, the menu data downloaded from the application servers to the user interface servers were transmitted within an XML document. As those skilled in the art will appreciate, this menu data may be transmitted in any appropriate format from the application servers to the user interface server. For example, this menu data may be transmitted in EJB (Enterprise JavaBeans) format. Since these formats are standard formats, a further description of them will be omitted.

In the above embodiment, both the request handling unit and the response handling unit could call one of the

common functions run by the common functions processor. In an alternative embodiment, only the request handling unit may be able to call the common functions. In this case, if an application server wishes to call one of the common functions, then it would have to transmit an appropriate request for the common function via the user set top box. This can easily be done using conventional web redirect techniques.

In the above embodiment, the management and billing server was responsible for monitoring the user requests that were made by all of the users from the data stored within the user request log of the user interface servers. It then used this information to adapt user profiles stored within the database. As those skilled in the art will appreciate, this task may be performed by a separate global operations controller (not shown) or it may be done individually by each of the application servers. For example, each of the application servers may be arranged to monitor the statistics relevant to the services offered by that application server. Each application server can then build a profile of each user that is relevant to that application server.

In the above embodiment, the user interface main menu had four menu options: a TVspace option, a Videspace option, a Yourspace option and an Openspace option. As those skilled in the art will appreciate other menu options [illegible]

service, which could be referred to as "Phonespace". Therefore, the personalised user interface of the user interface can be laid out in any number of logical sections depending on the number of different entertainment and activity types available in the system.

In the above embodiment, various application servers were described providing various television services to the users of the system. As those skilled in the art will appreciate, the various services that are available are described by way of illustration and should not be construed as limiting in any way. For example, in addition to the applications described above, the system may provide a time-shifted TV service in which programmes may be automatically recorded for users so that they can watch programmes after they have been broadcast.

In the above embodiment, each menu page was personalised for delivery to the user. This personalisation included personalised data received from the application servers as well as personalisation to include the users's name and to change the background colour of the menu screen in accordance with the user's preferences. In addition to these personalisations, the menu screen may also be personalised in terms of a language used, font used, the format of the date and time displayed etc. The personalisation may also include personalised advertising targeted to the user in accordance with their user profile. For example, by analysing a user's viewing habits and system usage, the system may determine that

the user likes action movies. Accordingly, the advertising may be targeted to products relating to such action movies.

In the above embodiment, a menu system has been described via which a user can gain access to various services provided by a number of remote application servers. In one embodiment, the user interface server preferably includes a help menu screen via which the user can access video help for each service and the overall operation of the system.

As those skilled in the art will appreciate, the client devices, the user interface servers and the application servers may be provided as hardware units or as a mixture of hardware and software components. If programmable computer devices are used as the basis for these components, the servers are preferably Microsoft NT servers, Linux Intel servers, Sun Solaris servers, Compac Alpha servers, IBM or HP servers or the like. Where user set top boxes are provided, these are preferably manufactured by Scientific Atlanta, Motorola, AT&T or Philips. If the user device is formed by a personal computer, then this is preferably a Pentium-based computer manufactured by, for example, Dell Computer Corporation, IBM or Toshiba and is connected either to a television or to another display device. The software used to control the servers to carry out the functions [illegible]

stored in compiled format, in uncompiled format or in any format intermediate the two. The software may be provided on a carrier such as a CD-ROM or the like or it may be downloaded over a data network such as the Internet.

In the above embodiment, various caches were provided both in the user interface server and in the application servers. These caches were provided in order to try to reduce the processing burden on either the application server or on the database. As those skilled in the art will appreciate, the caching performed in the above embodiment is not essential. One or more of the caches that are used may be omitted. For example, the XML cache within one or more of the user interface servers may be omitted leaving just the HTML cache and the XSLT cache. Additionally, a menu cache may be provided locally within each user device to store menu pages previously downloaded from the user interface server. In this case, the user device can check its local cache before transmitting a request for a next menu page. In this way, the number of requests transmitted to the user interface servers can also be minimised.

In the above embodiment, a variable swapping technique was used to swap in user personalisations into the menu pages that were generated within the user interface server. This technique was also used to swap in machine data for each of the user interface servers. As those skilled in the art will appreciate, this is not

essential. The menu pages that are generated may be generated for each specific user and for each user interface server. However, the use of these variable swapping techniques are preferred because it increases the effectiveness of the caching being employed because of the more generic nature of the cached menu pages. Further, if a variable swapping technique is used, it is not essential to use the hash delimiters to identify the placeholder entries. Any suitable character which is not a control character for HTML or the style sheets could be used.

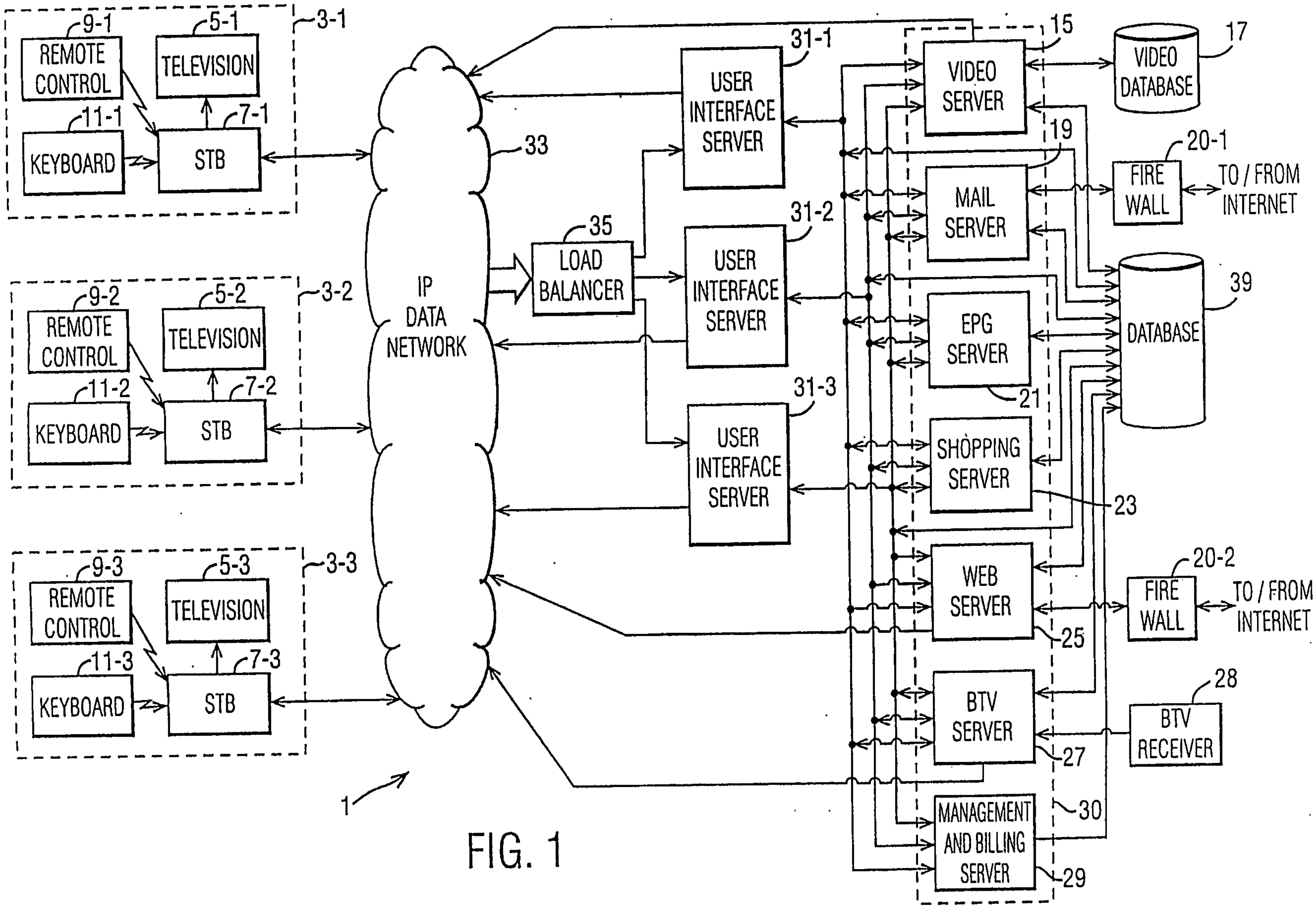

REMOTE CONTROL
TELEVISION
KEYBOARD
STB
IP DATA NETWORK
LOAD BALANCER
USER INTERFACE SERVER
VIDEO SERVER
VIDEO DATABASE
MAIL SERVER
FIRE WALL
TO / FROM INTERNET
DATABASE
EPG SERVER
SHOPPING SERVER
WEB SERVER
BTV SERVER
BTV RECEIVER
MANAGEMENT AND BILLING SERVER
3-1
3-2
3-3
5-1
5-2
5-3
7-1
7-2
7-3
9-1
9-2
9-3
11-1
11-2
11-3
1
15
17
19
20-1
20-2
21
23
25
27
28
29
30
31-1
31-2
31-3
33
35
39

FIG. 1

FIG. 2a

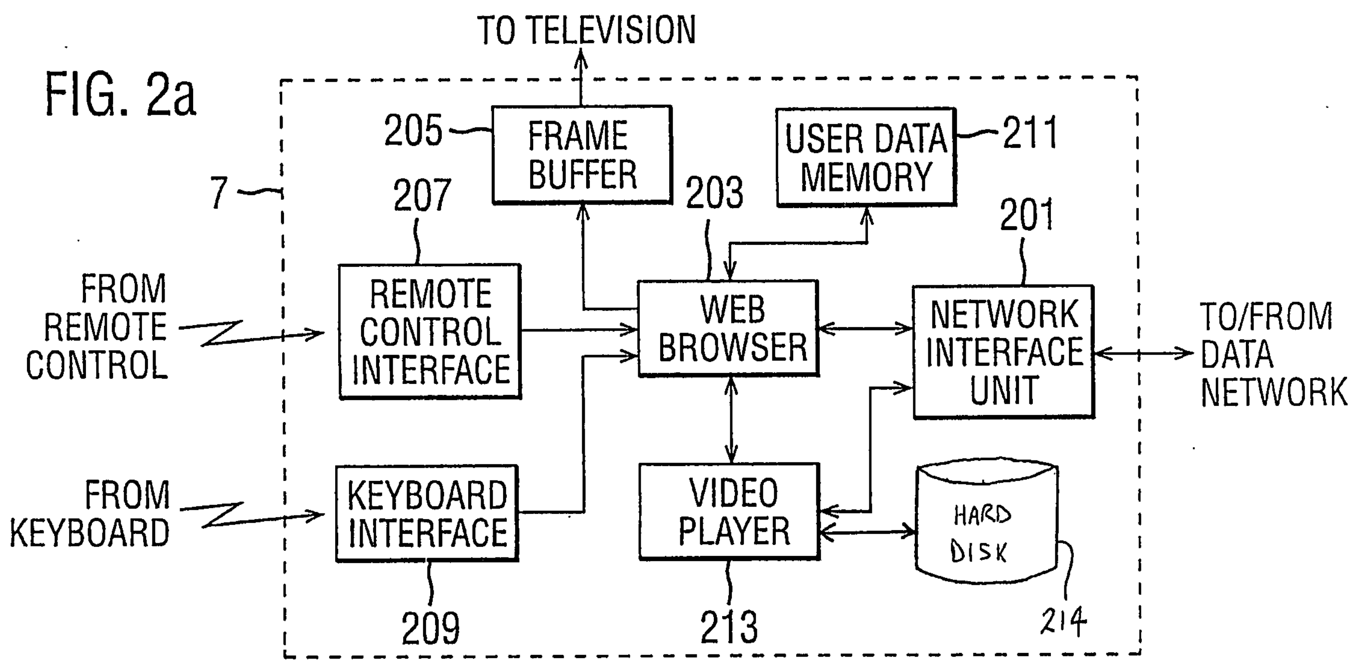

TO TELEVISION
205
FRAME BUFFER
USER DATA MEMORY
211
7
207
203
201
FROM REMOTE CONTROL
REMOTE CONTROL INTERFACE
WEB BROWSER
NETWORK INTERFACE UNIT
TO/FROM DATA NETWORK
FROM KEYBOARD
KEYBOARD INTERFACE
VIDEO PLAYER
HARD DISK
209
213
214

FIG. 2b

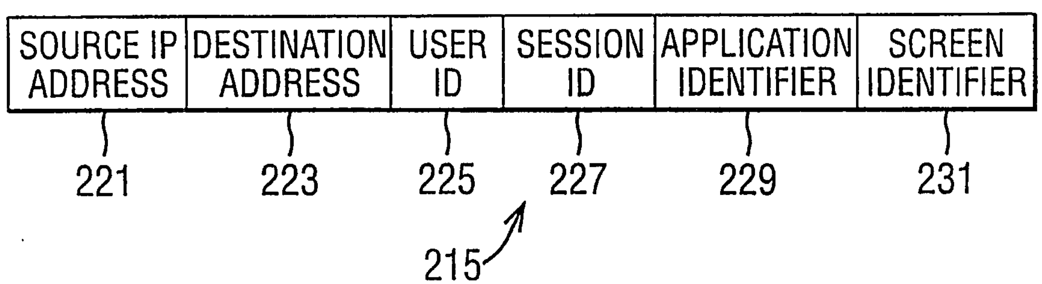
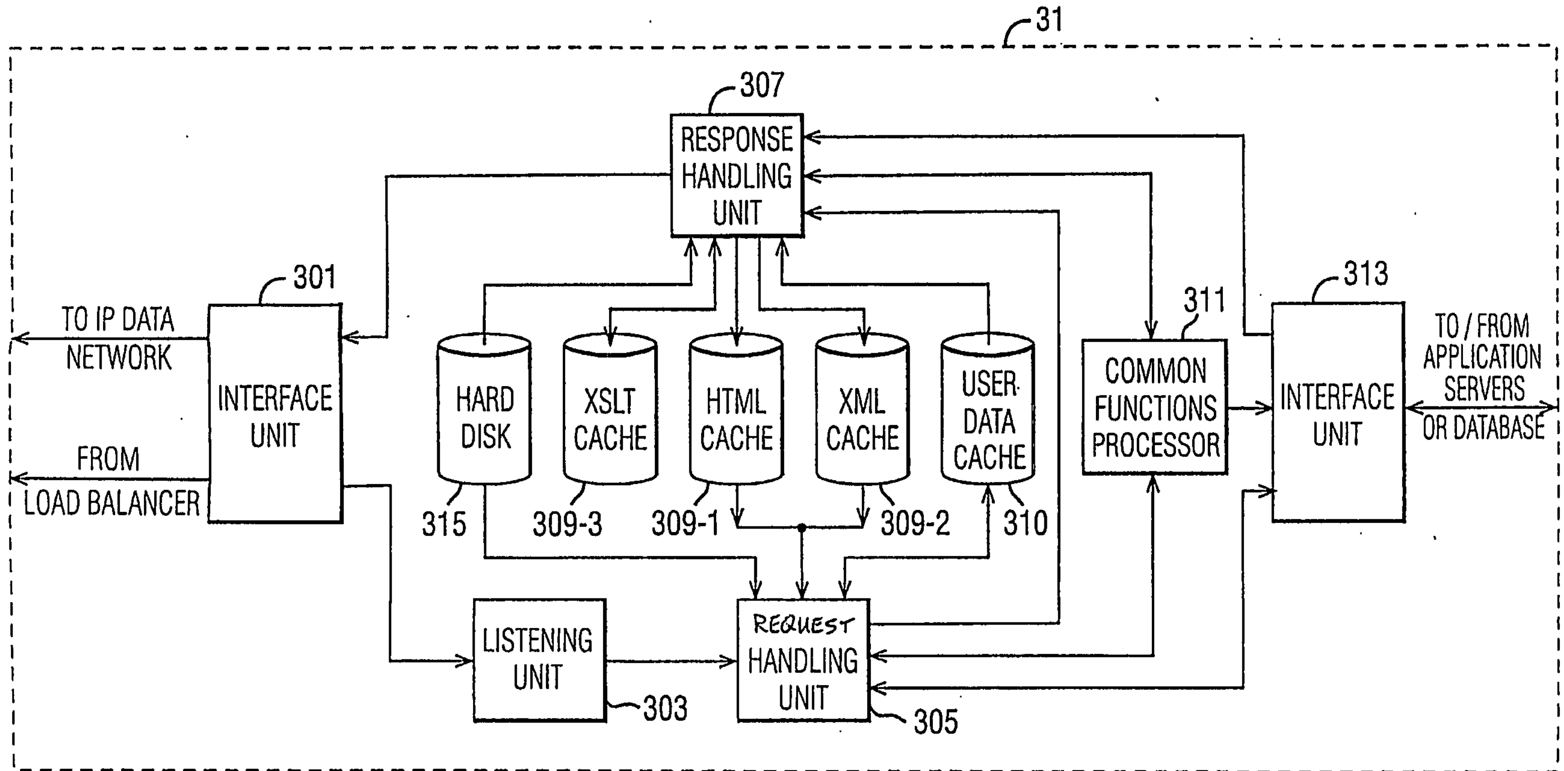

SOURCE IP ADDRESS
DESTINATION ADDRESS
USER ID
SESSION ID
APPLICATION IDENTIFIER
SCREEN IDENTIFIER
221
223
225
227
229
231
215

FIG. 3



31
307
RESPONSE HANDLING UNIT
301
TO IP DATA NETWORK
FROM LOAD BALANCER
INTERFACE UNIT
HARD DISK
XSLT CACHE
HTML CACHE
XML CACHE
USER DATA CACHE
315
309-3
309-1
309-2
310
311
COMMON FUNCTIONS PROCESSOR
313
INTERFACE UNIT
TO / FROM APPLICATION SERVERS OR DATABASE
LISTENING UNIT
303
REQUEST HANDLING UNIT
305

FIG. 4

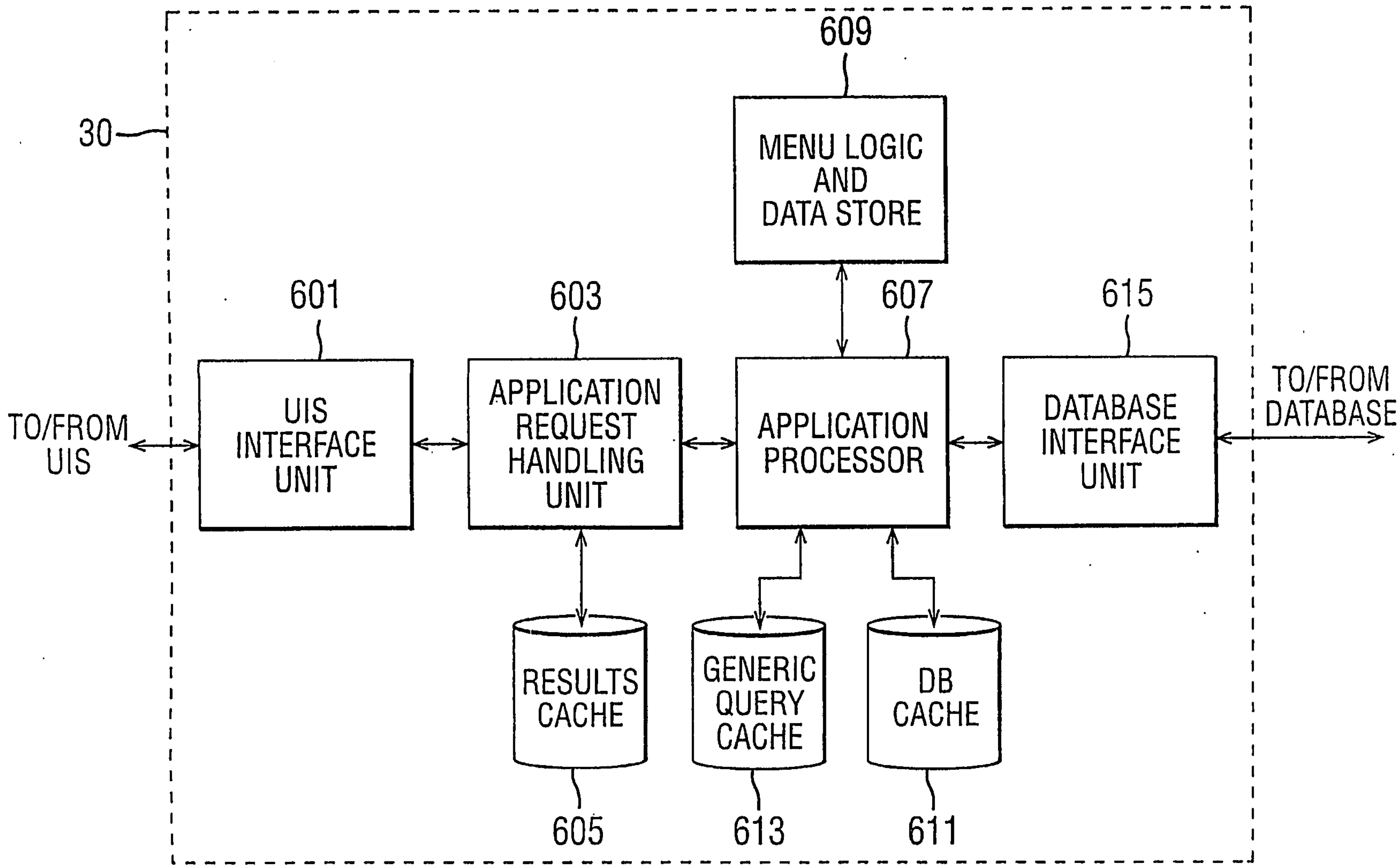

30
TO/FROM UIS
601
UIS INTERFACE UNIT
603
APPLICATION REQUEST HANDLING UNIT
609
MENU LOGIC AND DATA STORE
607
APPLICATION PROCESSOR
615
DATABASE INTERFACE UNIT
TO/FROM DATABASE
RESULTS CACHE
605
GENERIC QUERY CACHE
613
DB CACHE
611

FIG. 5

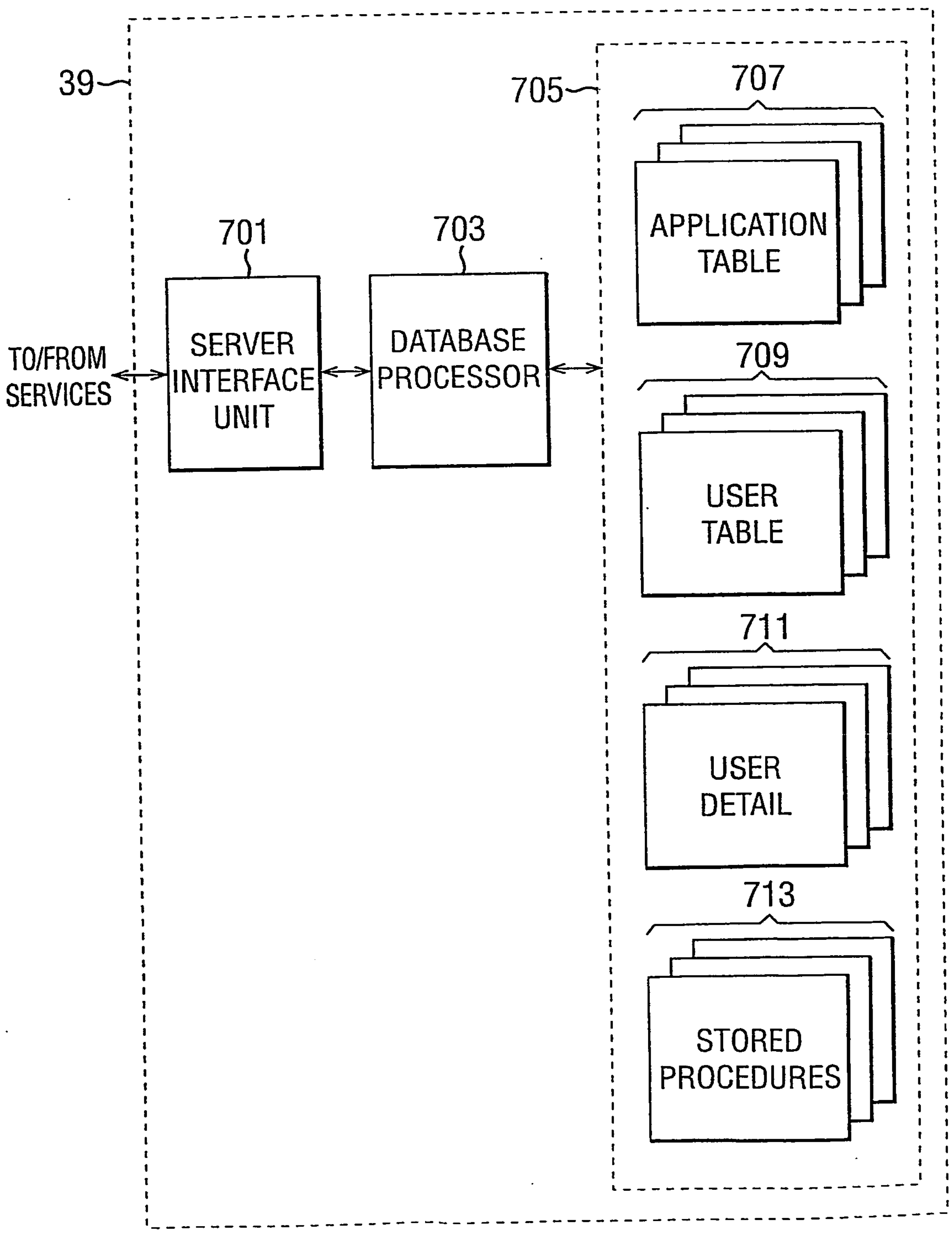
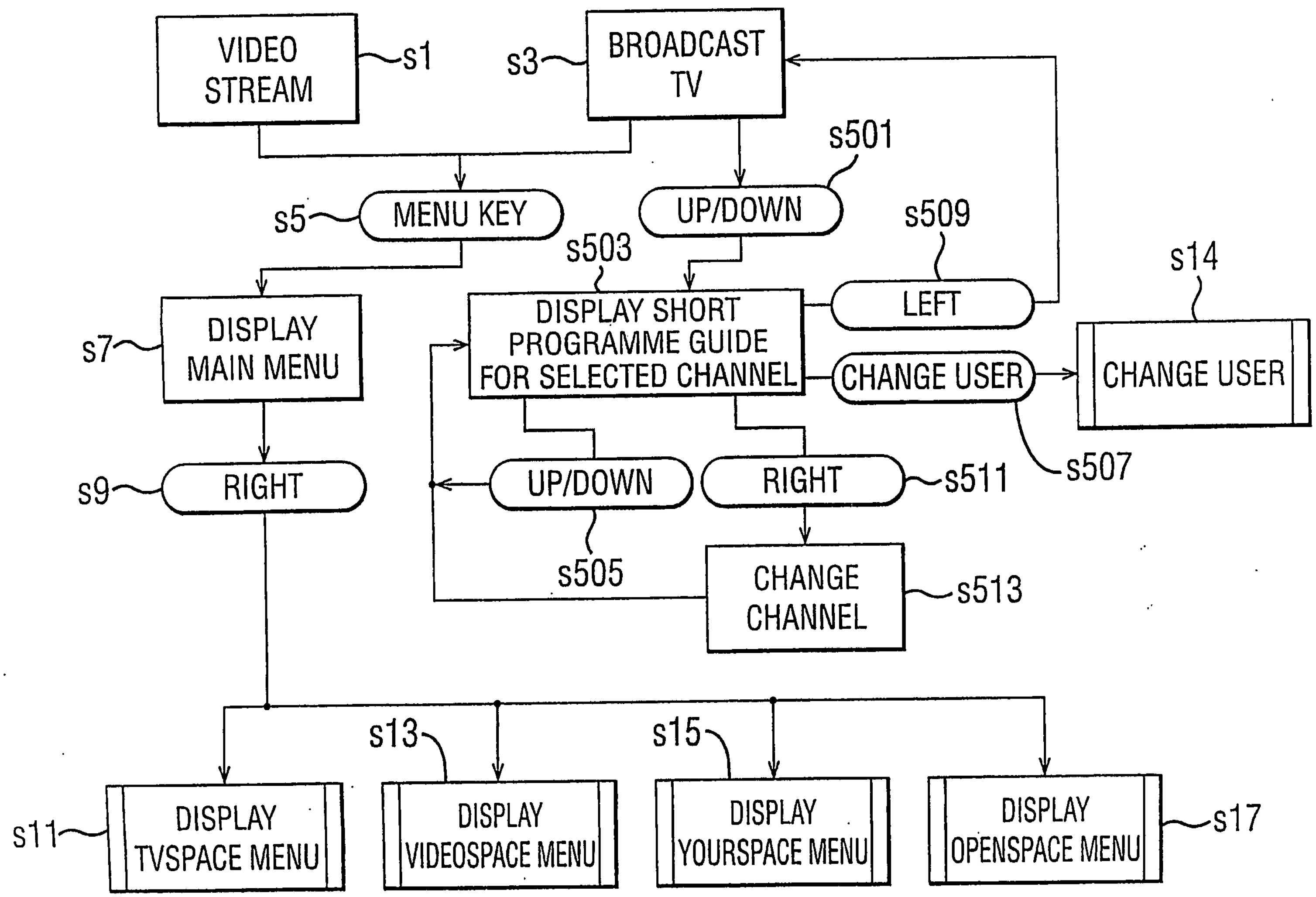

39
705
707
701
703
APPLICATION TABLE
TO/FROM SERVICES
SERVER INTERFACE UNIT
DATABASE PROCESSOR
709
USER TABLE
711
USER DETAIL
713
STORED PROCEDURES

FIG. 6



VIDEO STREAM
s1
BROADCAST TV
s3
MENU KEY
s5
UP/DOWN
s501
s503
DISPLAY SHORT PROGRAMME GUIDE FOR SELECTED CHANNEL
LEFT
s509
CHANGE USER
s507
CHANGE USER
s14
DISPLAY MAIN MENU
s7
RIGHT
s9
UP/DOWN
s505
RIGHT
s511
CHANGE CHANNEL
s513
DISPLAY TVSPACE MENU
s11
DISPLAY VIDEOSPACE MENU
s13
DISPLAY YOURSPACE MENU
s15
DISPLAY OPENSPACE MENU
s17

FIG. 7

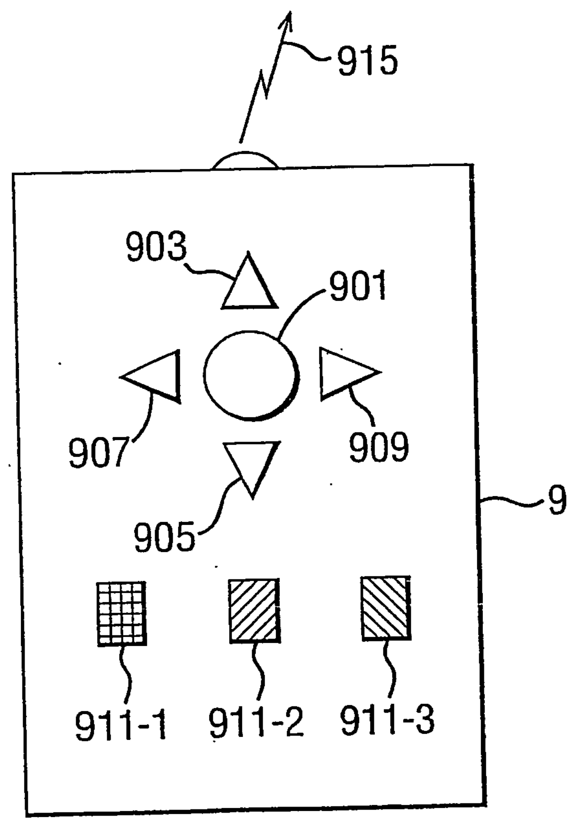
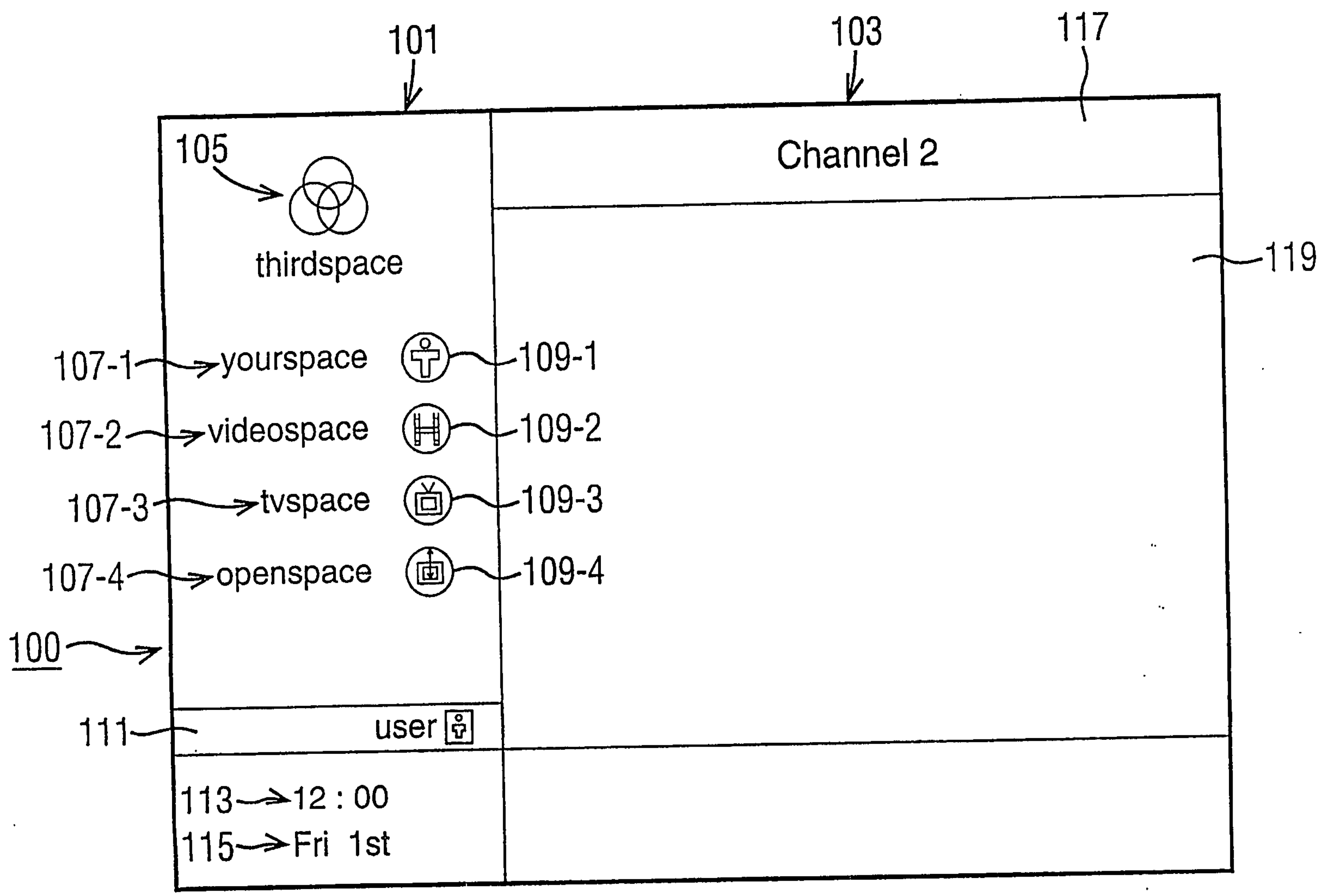
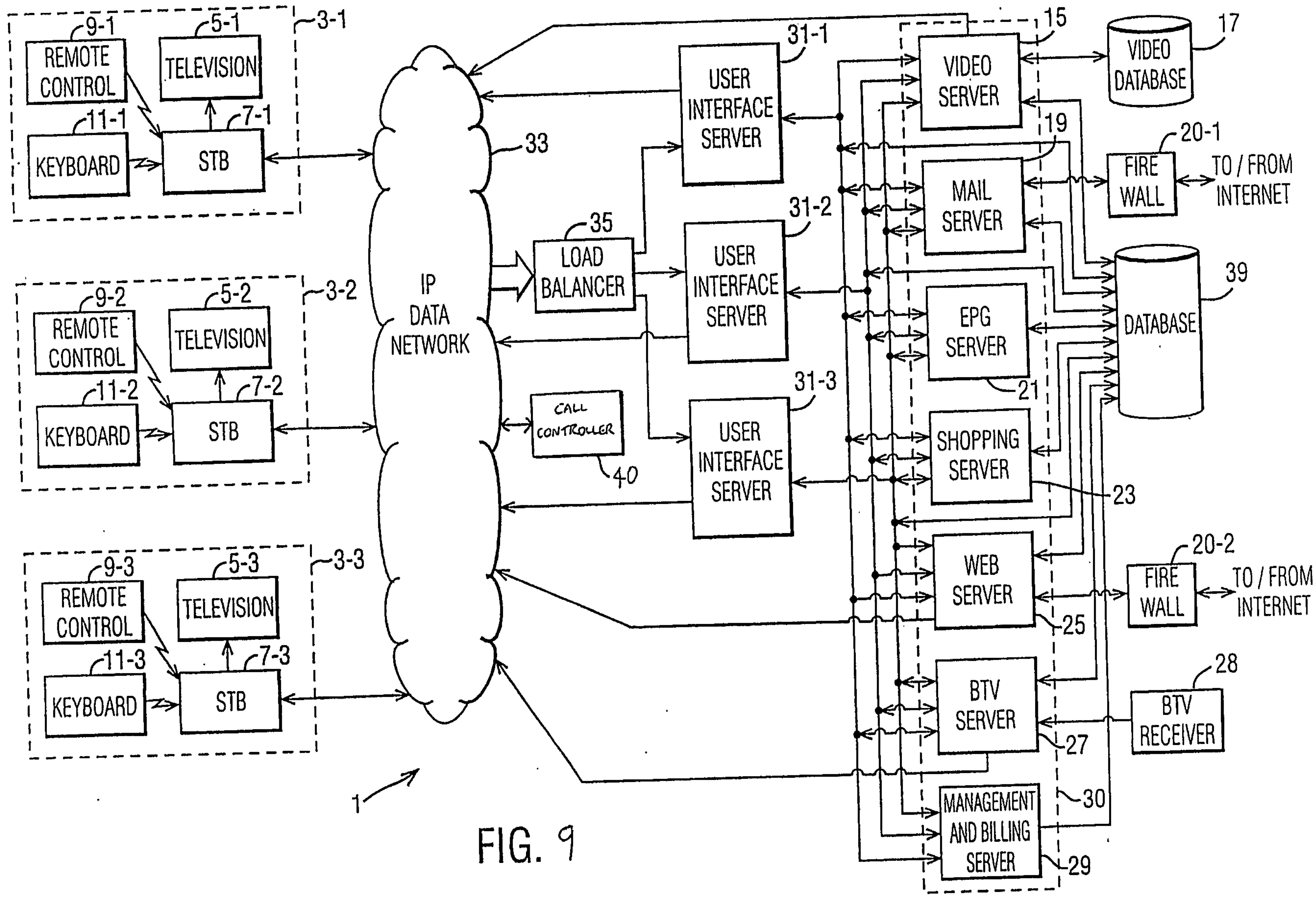


FIG. 8


101
103
117
105
thirdspace
Channel 2
119
107-1
yourspace
109-1
107-2
videospace
109-2
107-3
tvspace
109-3
107-4
openspace
109-4
100
111
user
113
12 : 00
115
Fri 1st



REMOTE CONTROL
TELEVISION
KEYBOARD
STB
9-1
5-1
11-1
7-1
3-1
REMOTE CONTROL
TELEVISION
KEYBOARD
STB
9-2
5-2
11-2
7-2
3-2
REMOTE CONTROL
TELEVISION
KEYBOARD
STB
9-3
5-3
11-3
7-3
3-3
IP DATA NETWORK
33
LOAD BALANCER
35
CALL CONTROLLER
40
USER INTERFACE SERVER
31-1
USER INTERFACE SERVER
31-2
USER INTERFACE SERVER
31-3
VIDEO SERVER
15
MAIL SERVER
19
EPG SERVER
21
SHOPPING SERVER
23
WEB SERVER
25
BTV SERVER
27
MANAGEMENT AND BILLING SERVER
29
30
VIDEO DATABASE
17
FIRE WALL
20-1
TO / FROM INTERNET
DATABASE
39
FIRE WALL
20-2
TO / FROM INTERNET
BTV RECEIVER
28
1

FIG. 9